

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-032182

(43)Date of publication of application : 31.01.2002

(51)Int.Cl.

G06F 3/02
G06F 3/00

(21)Application number : 2000-219576

(71)Applicant : RICOH CO LTD

(22)Date of filing : 19.07.2000

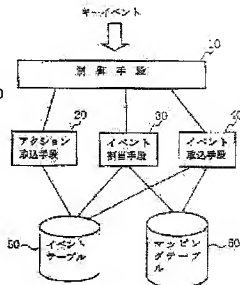
(72)Inventor : SHIMADA NAOKI

(54) KEY ALLOCATING SYSTEM AND KEY ALLOCATING METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a key allocating system which a beginner can easily handle and in which customization is easy.

SOLUTION: The system is provided with an event table 50 which correspondingly holds the action of the input operation of a key or a mouse and an event name causing the action, a mapping table 60 which correspondingly holds the event name and a series of key operations for causing the event, an action take-in means 20 taking in a series of key operations, giving the event names and registering them to the event table 50, and an event allocation means 30 which correspondingly stores the event names registered in the action take-in means 20 and a series of operations which actually cause the event in the mapping table 60. When the event is designated by the key operation, a series of key operations corresponding to the events registered in the mapping table 60 and the event table 50 are performed.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-32182
(P2002-32182A)

(43) 公開日 平成14年1月31日 (2002.1.31)

(51) Int. Cl. ⁷	識別記号	F I	キーワード (参考)
G 0 6 F 3/02	3 6 0	G 0 6 F 3/02	3 6 0 G 5 B 0 2 0
	3 1 0		3 1 0 K 5 E 5 0 1
3/00	6 1 0	3/00	6 1 0

審査請求 未請求 請求項の数6 O L (全 8 頁)

(21) 出願番号 特願2000-219578(P2000-219576)

(22) 出願日 平成12年7月19日 (2000.7.19)

(71) 出願人 000008747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72) 発明者 島田 直樹

東京都大田区中馬込1丁目3番6号 株式

会社リコー内

Fターム(参考) 5B020 CC07 CC12 G022 G025

5E501 BA05 CB02 CB09 DA11 DA14

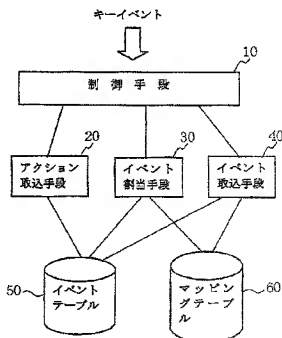
DA15 EA05 FA13

(54) 【発明の名称】 キー割当てシステムおよびキー割当て方法

(57) 【要約】

【課題】 初心者にも扱いやすく、カスタマイズの簡単なキー割当てシステムを提供する。

【解決手段】 本発明は、キーまたはマウス等の入力操作のアクションとそのアクションを引き起こすイベント名とを対応付けて保持するイベントテーブル50と、これのイベント名とそのイベントを引き起こすための一連のキー操作とを対応させて保持するマッピングテーブル60と、前記一連のキー操作を取り込んでイベント名を付けてイベントテーブル50へ登録するアクション取込手段20と、このアクション取込手段20で登録されたイベント名に対して実際のイベントを引き起こす一連の操作とを対応させてマッピングテーブル60へ格納するイベント割当手段30とを備え、キー操作によりイベントが指定されたときにマッピングテーブル60とイベントテーブル50に登録されているこのイベントに対応する一連のキー操作を実行するように構成されている。



【特許請求の範囲】

【請求項1】 システムに用意されているキーに対する機能の割当て変更が可能なキー割当てシステムにおいて、キーまたはマウス等の入力操作のアクションとそのアクションを引き起こすイベント名とを対応付けて保持するイベントテーブルと、前記イベントテーブルのイベント名とそのイベントを引き起こすための一連のキー操作とを対応させて保持するマッピングテーブルと、前記一連のキー操作を取り込んでイベント名を付けて前記イベントテーブルへ登録するアクション取込手段と、前記アクション取込手段で登録されたイベント名に対して実際のイベントを引き起こす一連の操作とを対応させて前記マッピングテーブルへ格納するイベント割当て手段とを備え、キー操作によりイベントが指定されたときに前記マッピングテーブルと前記イベントテーブルに登録されているこのイベントに対応する一連のキー操作を実行するようにしたことを特徴とするキー割当てシステム。

【請求項2】 請求項1記載のキー割当てシステムにおいて、前記アクション取込手段は、取り込まれたキー操作を文字列に変換し、前記イベントテーブルへ保持させるようにしたことを特徴とするキー割当てシステム。

【請求項3】 請求項1または請求項2記載のキー割当てシステムにおいて、取り込まれた一連のキー操作に前記イベントテーブルのイベント名と対応付けて前記マッピングテーブルへ登録するイベント取込手段を備え、ユーザがイベントを引き起こすトリガーを指定できるようにしたことを特徴とするキー割当てシステム。

【請求項4】 システムに用意されているキーに対する機能の割当て変更が可能なキー割当てシステムにおいて、キーまたはマウス等の入力操作のアクションとそのアクションを引起こすイベント名とを対応付けて保持するイベントテーブルと、前記イベントテーブルのイベント名とそのイベントを引き起こすための一連のキー操作とを対応させて保持するマッピングテーブルとを有し、前記一連のキー操作を取り込んでイベント名を付けて前記イベントテーブルへ登録し、このイベント名に対して実際のイベントを引き起こす一連の操作とを対応させて前記マッピングテーブルへ格納することによって、キー操作によりイベントが指定されたときに前記マッピングテーブルと前記イベントテーブルに登録されているこのイベントに対応する一連のキー操作を実行するようにしたことを特徴とするキー割当て方法。

【請求項5】 請求項4記載のキー割当て方法において、前記アクション取込手段は、取り込まれたキー操作を文字列に変換し、前記イベントテーブルへ保持させるようにしたことを特徴とするキー割当て方法。

【請求項6】 請求項4または請求項5記載のキー割当て方法において、取り込まれた一連のキー操作に前記イベントテーブルのイベント名と対応付けて前記マッピングテーブルへ登録し、ユーザがイベントを引き起こす

トリガーを指定できるようにしたことを特徴とするキー割当て方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、キー割当てシステムおよびキー割当て方法に関し、特に、ワープロまたはパソコン作業の機能操作に対しキーを割当てて技術に関する。

【0002】

【従来の技術】 従来より、例えば、かな漢字変換装置には、カタカナ変換や半角変換等のかな漢字の種々な機能がキーに設定されており、ユーザはそのキーを押下することにより、そのキーに設定されている機能を簡単に実行することが可能となっている。これらのキーは、かな漢字変換装置の環境設定プログラムにより、デフォルトで設定されているキーをユーザの好みに応じて自由に変更することが可能なものである。また、キーボードマクロのように、あるキーに対し特定のキー操作列を文字列として定義しておき、実行に際しては、このキーを入力することによって定義されたマクロが実行されるようにしたシステムもある。

【0003】

【発明が解決しようとする課題】 上述したように、従来の技術では、あるキー操作に対して、キーを割当てて場合には、与えられたテーブルから選択したり、また、マクロを設定したりしなければならないので初心者には扱いづらく、キーカスタマイズを行えないという問題がある。本発明の課題は、このような問題を解決することにより、初心者にも扱いやすく、カスタマイズの簡単なキー割当てシステムおよびキー割当て方法を提供することによって、コンピュータにおける作業効率を向上させる。

【0004】

【課題を解決するための手段】 前記課題を解決するために、請求項1に記載の発明は、システムに用意されているキーに対する機能の割当て変更が可能なキー割当てシステムにおいて、キーまたはマウス等の入力操作のアクションとそのアクションを引き起こすイベント名とを対応付けて保持するイベントテーブルと、前記イベントテーブルのイベント名とそのイベントを引き起こすための一連のキー操作とを対応させて保持するマッピングテーブルと、前記一連のキー操作を取り込んでイベント名を付けて前記イベントテーブルへ登録するアクション取込手段と、前記アクション取込手段で登録されたイベント名に対して実際のイベントを引き起こす一連の操作とを対応させて前記マッピングテーブルへ格納するイベント割当て手段とを備え、キー操作によりイベントが指定されたときに前記マッピングテーブルと前記イベントテーブルに登録されているこのイベントに対応する一連のキー操作を実行するようにしたことを特徴とする。請求項2

に記載の発明は、請求項1記載のキー割当てシステムにおいて、前記アクション取込手段は、取り込まれたキー操作を文字列に変換し、前記イベントテーブルへ保持させるようにしたことを特徴とする。請求項3に記載の発明は、請求項1または請求項2記載のキー割当てシステムにおいて、取り込まれた一連のキー操作に前記イベントテーブルのイベント名と対応付けて前記マッピングテーブルへ登録するイベント取込手段を備え、ユーザがイベントを引き起こすトリガーを指定できるようにしたことを特徴とする。請求項4に記載の発明は、システムに用意されているキーに対する機能の割当て変更が可能なキー割当てシステムにおいて、キーやマウス等の入力操作のアクションとそのアクションを引き起こすイベント名とを対応付けて保持するイベントテーブルと、前記イベントテーブルのイベント名とそのイベントを引き起こすための一連のキー操作とを対応させて保持するマッピングテーブルとを有し、前記一連のキー操作を取り込んでイベント名を付けて前記イベントテーブルへ登録し、このイベント名に対して実際のイベントを引き起こす一連の操作とを対応させて前記マッピングテーブルへ格納することによって、キー操作によりイベントが指定されたときに前記マッピングテーブルの前記イベントテーブルに登録されているこのイベントに対応する一連のキー操作を実行するようにしたことを特徴とする。請求項5に記載の発明は、請求項4記載のキー割当て方法において、前記アクション取込手段は、取り込まれたキー操作を文字列に変換し、前記イベントテーブルへ保持させるようにしたことを特徴とする。請求項6に記載の発明は、請求項4または請求項5記載のキー割当て方法において、取り込まれた一連のキー操作に前記イベントテーブルのイベント名と対応付けて前記マッピングテーブルへ登録して、ユーザがイベントを引き起こすトリガーを指定できるようにしたことを特徴とするキー割当て方法。

【0005】

【発明の実施の形態】以下、本発明の1つの実施の形態を図面に基づいて詳細に説明する。図1は、本発明の1つの実施の形態に係るキー割当てシステムを説明するためのブロック図である。図1に示すように、本発明の1つの実施の形態に係るキー割当てシステムは、制御手段10と、アクション取込手段20と、イベント割当て手段30と、イベント取込手段40と、イベントテーブル50およびマッピングテーブル60とから構成されている。制御手段10は、キーボード操作やマウス操作などの入力と他の機能手段等を制御する。アクション取込手段20は、ある作業を実現するキーボード操作やマウス操作の系列をそのまま取り込み、その系列を文字列に変換して、一時的にバッファへ保持し、バッファの内容とそのアクションを引き起こすイベントの名前を対応付けてイベントテーブル50へ格納する。ここで、イベントが

起こったときのアクションを文字列形式としたことにより、次のような利点が考えられる。従来技術（例えば、特開平5-274078号公報または特開平9-190260号公報等参照）では、イベントを信号を使って直接記憶したり、呼び出したりしていたが、本発明のように文字列として扱うことにより、ユーザが視覚的にイベント内容を確認することができる。また、ユーザが自分でイベントを編集することも可能となる。さらに、はじめからキー操作をユーザが簡単な文法を使って書くことも可能である。文法には、キー操作だけでなくWS、W10（「W」の後の数字だけ待つ）という動作等も用意する。これにより、アプリケーションが起動する時間をあらかじめ計算して待ち、起動後に次のキー操作を行うこともできる。イベント割当て手段30は、アクション取込手段20で一時的にバッファに保持されている文字列に与え、イベントテーブル50に登録されたイベント名をユーザに選択させて、この選択されたイベント名に対応する実際の入力されるイベントとを対応させてマッピングテーブル60へ登録する。イベント取込手段40は、イベントテーブル50のイベント名に実際の入力するイベントをユーザが定義したいときに動作させ、イベントとして定義するキーボード操作およびマウス操作の系列を文字列として取込み、その系列に対してイベント名を対応させてマッピングテーブル60へ登録する。イベントテーブル50は、ある一連のキーボード操作やマウスの操作を文字列に変換して、その文字列に対してイベント名を対応させて管理するテーブルである。【0006】図2は、このイベントテーブル50のデータ構造図を示している。イベントテーブル50は、イベント名とアクション列とからなる。イベント名はアクション列に定義された文字列に対して与える名前である。アクション列はイベントに対し、どのようなアクションを引き起こすかを指示する文字列である。例えば、「上町に定例会議が無事終了したことを会議結果報告と言う題名のメールで報告する。」という場合を考える。この場合、大きく分けて、以下の4つの動作を必要とする。

- (1) メーリングソフトを立ちあげる。
- (2) メールの新規作成を行ない、宛先、題名および本文を書く。
- (3) 送信する。
- (4) メーリングソフトを閉じる。

メーリングソフトの実行ファイル名が、mail.exeであり、ファイル名を指定して実行を行うことができる。すなわち、「W10」、「Ctrl+F」、「Esc」、「r」、「m」、「a」、「i」、「l」、「/」、「e」、「x」、「e」、「Enter」の操作でメーリングソフトを呼び出せる。これらの意味は次のとおりである。「W10」は、プログラムの起動に7〜8秒かかることと仮定して、10秒待つ。「Ctrl+F」は、スタートメニューを表示する。「r」は、スタートメニューから

らファイル名を指定して実行を選択する。「m」「a」「l」「i」「e」「x」「e」「e」「Enter」は、メーリングソフトを実行する。次に、メールの新規作成をアクセスキーで行なえること、また、題名、宛先、本文をタブキー等で動かせること、及び、日本語入力が替がキー操作で行なえるとした場合、「Alt+m+a+s」、「e」、「a」、「l」、「i」、「e」、「x」、「e」、「Enter」、「Tab」、「k」、「a」、「s」、「g」、「i」、「k」、「e」、「k」、「a」、「h」、「o」、「u」、「変換」、「Tab」、「e」、「a」、「s」、「g」、「i」、「k」、「e」、「k」、「a」、「h」、「o」、「u」、「変換」の操作で実行できる。これらの意味は次のとおりである。ただし、漢字を使う際には、一回で変換できるようにすることが望ましい。「Alt+m+a+s」はアクセスキーである。「e」、「a」、「l」、「i」、「e」、「x」、「e」、「Enter」はメールアドレスの設定である。「Alt+a+半角」は日本語入力に設定である。「Tab」は題名にカーソルを動かすのである。「k」、「a」、「s」、「g」、「i」、「k」、「e」、「k」、「a」、「h」、「o」、「u」、「変換」、「Tab」、「e」、「a」、「s」、「g」、「i」、「k」、「e」、「k」、「a」、「h」、「o」、「u」、「変換」は題名の入力である。「Tab」は本文にカーソルを動かすのである。「e」、「a」、「l」、「i」、「e」、「x」、「e」、「Enter」は本文の入力である。

【0007】次に、送信もアクセスキーで行なえることすれば、「Alt+m+a+p」の操作で実行できる。次に、アプリケーションソフトの終了もアクセスキーで行なえることすれば、「Alt+a+f+s」の操作で実行できる。このような一連の操作をテキスト化しただけでは、ワンタッチで、メーリングソフトを起動して定型的なメールを書き、相手に送信してメールを閉じるといったことも可能になる。また、テキストを編集すれば、すべてを書き直すことなく、題名の部分や本文の部分を変えるだけで、別の内容のメールを送るように設定することもできる。上記のような「会報結果報告のメールを出す」例では、イベントテーブル50の内容は、「イベント名は会報結果報告イベントであり、アクション列は上述した4つのアクションを1つに連結した文字列である。」のようなエントリが設定される。また、「イベント名は再起動イベントであり、アクション列はAlt+Esc、u、l、Enterである。」のような例も考えられる。このイベントテーブル50は、メモリやハードディスク等の記憶装置に格納されるが、キーボードやマウス等の入力手段が実行中のときはメモリへ常駐するのがよい。マッピングテーブル60は、イベントを引き起こすための実際の一連のキーボード操作やマウス操作等を文字列に変換し、この文字列とイベントテーブル50に保持されたイベント名とを対応付けて管理するテーブルである。

【0008】図3は、このマッピングテーブル60のデータ構造図を示している。図3に示すように、マッピングテーブル60は、「イベント名がイベントテーブル50のイベント名であり、アクション列がイベント名で表されるイベントを実際に引き起こすための一連のキーボード操作やマウス操作等を文字列に変換したもののあ

る。フォーカスのあたっているオブジェクトに対して、ダブルクリックイベントを呼び出す場合には、「イベント名がダブルクリックであり、アクション列がl、b、r、c、i、l、i、l、c、kである。」と設定される。また、マシンの再起動を行なうイベントを呼び出す場合には、「イベント名が再起動であり、アクション列がl、r、e、b、o、o、oである。」に設定される。また、いつもの友達にランチを食べに行こうというメールを出すイベントを呼び出す場合には、「イベント名がランチのメールであり、アクション列がl、l、u、b、n、c、hである。」に設定される。このように、ユーザが覚えやすいように命名することができる。このマッピングテーブル60は、メモリやハードディスク等の記憶装置に格納され、キーボードやマウス等の入力手段が実行中のときはメモリへ常駐するのがよい。

【0009】次に、キー割当てシステムの動作概要を説明する。キー割当てシステムを動作させ、例えば、図4に示したようなユーザーインターフェースを備えた画面を表示する。まず、ユーザが「記憶開始ボタン」を押すと、制御手段10はアクション取手段20を動作させ、これ以降のキー操作をバッファへ一時的に書き込む。この書き込みは、以下の文法に従ってバッファへ書き込む。キーボードのキーダウンを感知したら、その番号からどのキーが押されたかを感知し、キー毎に固有の文字列を定義しておき、番号をその文字列に変換してバッファへ追加していく。このバッファの内容は図4の画面に表示される。例えば、aというキーが押されたときにはバッファに文字列「a」を追加する。また、CtrlキーやAltキーが押された時には、文字列「Ctrl」、「Alt」のようにすべてのキーを文字列化してバッファへ追加する。さらに、1つのキーがダウンしている時に、別のキーがダウンされた場合は、文字列「d」を加え、すべてのキーがアップされた状態になったら文字列「u」で区切るようにしてバッファへ追加する。例えば、「スタートメニュー」を表示し(Ctrl+Esc)、「ファイル名を指定して実行(r)」を選択し、テキストボックスに「a.exe」と入力しOKボタンを押す。」という一連のキー操作は、「Ctrl+Esc、r、a、.、e、x、e、Enter」のようにバッファに記憶される。このようなある仕事に対応させる一連のキー操作をバッファに記憶させた後、「記憶終了ボタン」を押すと、このような一連のキー操作の取り込みを中止し、このバッファの内容に該当イベント名をユーザに付けさせて、イベントテーブル50へ格納する。

【0010】次に、ユーザが「イベント一覧ボタン」を押すと、制御手段10はイベント割当て手段30を起動させる。イベント割当て手段30は、イベント一覧を表示させ、ユーザは先に記憶したバッファ中の文字列に与えたイベント名に実際のイベントをこの一覧表から選択する。選択されたイベントとイベント名とを対応させてマ

ッピングテーブル60へ登録する。例えば、右ダブルクリック、左トリプルクリック、右トリプルクリック、F1キー押下、F2キー押下...等およびユーザが割当てたイベントがイベント一覧に表示される。この中からユーザが“右ダブルクリック”を選択すると、このイベントとバッファの内容に対するイベント名とがマッピングテーブル60へ格納される。本キー割当てシステムの動作が完了して以降、キーボード操作やマウス操作が行われると、それらからの入力を取り扱う入力手段は、入力された信号に応じてマッピングテーブル60のイベント、即ち、アクション列に登録された文字列と同一の文字列が見つかったと、それに対応するイベント名をイベントテーブル50に探し、同じものが見つければ、そのアクション列を実行させる(図5参照)。例えば、上例のように、“右ダブルクリック”が行われると、A.exeが起動されるようになる。上述のイベント割当て手段30にて、イベント一覧を表示させたとき、“ユーザによる設定”を選択すると、イベント取込手段40が起動される。このイベント取込手段40は、アクション取込手段20と同様の形式で、キーボード操作やマウス操作を文字列へ変換して一時的にバッファへ取り込んだ文字列とイベント名とを対応付けてマッピングテーブル60へ登録する。例えば、いつもの友達に“ランチを食べに行こうよ”と誘うメールを出す場合などは、“!hunch”という文字列をキーイベントとして登録しておくことで、メールを出すようにできる。また、キーイベントをパスワード的に利用することにより、パスワードを入力することでプライベートなファイルを開くことも可能である。例えば、一つのマシンを他人が使う可能性があるとき、他人にはアクセスされないファイルがあった場合、見つかりにくいパス、または、見つかりにくいファイル名を設定しておけば、見つかることはめったにないであろうが、ユーザ自身もそのファイル名やパスを忘れてしまったりする恐れがある。そこで、自分でも忘れないような(自分しか知らないような)イベントを設定しておく、このイベントを起こさせることによってファイルを開き、パスワード入力でファイルアクセスをする形式が実現できる。例えば、イベント名「毎日自分の体重の推移をまとめたファイルを開く」。およびアクション列「!Secret」やイベント名「借金の返済計画をまとめたファイルを開く」。およびアクション列「!H10」などがあげられる。

【0011】次に、キー割当てシステムの処理方法を説明する。図6は、キー割当てシステムの処理の流れを説明するフローチャートである。この処理のユーザーインターフェースを図4の画面表示を使って説明する。まず、実行に先立ち一連のキー操作を取り込んでいる状態(以下、アクションモードという。)か、そうでないかを示すモードスイッチをクリアする(ステップS100)。キーボードおよびマウス等からの入力信号がない

かどうかをセンシングする(ステップS110)。このセンシングの結果、“記憶開始ボタン”、“記憶終了ボタン”、“イベント一覧ボタン”または“終了ボタン”等の“ボタン”からのイベントであるかどうかを調べる(ステップS120)。ボタンからの信号の時にはステップS130へ進む。そうでなければステップS160へ進む。“記憶開始ボタン”からのイベントかどうか調べる。そうであればステップS140へ進み、そうでなければステップS210へ進む。“記憶開始ボタン”からの信号の時には、キー操作列を文字列化して一時的に保持するバッファをクリアする(ステップS140)。以後、アクションモードとして設定し、キー操作列を取り込むようにして(ステップS150)、ステップS170へ進む。ステップS120で“ボタン”からのイベントでなかった場合、キー又はマウス等からの入力信号であるとする。このとき既にアクションモードに設定されているかを調べる(ステップS160)。まだ、アクションモードに設定されていない場合は、このキー操作を取り込まないので、次のイベントをセンシングするためにステップS110へ戻る。既にアクションモードになっているときには、ステップS170へ進む。入力されたキー信号を文字列に変換し、その文字列をさらに“”で囲んだ文字列にする。この文字列をバッファへ追加する(ステップS170)。また、バッファが空の状態でないときには、文字カンマ(,)を付けて上記の文字列をバッファに追加する。今処理しているキーと同時に別のキーも押されているかを調べる(ステップS180)。押されていればステップS190へ進み、押されておらずすべてのキーがアップの状態になっていれば、次のイベントセンシングのためにステップS110へ戻る。同時に、複数のキーが押されていれば、文字プラス(+)をバッファへ追加し(ステップS190)、その別のキー信号を文字列に変換し、その文字列をさらに“”で囲んだ文字列をバッファへ追加し(ステップS200)、別のキーが同時に押されているかを調べるためにステップS180へ戻る。

【0012】ステップS210で押されたボタンが“記憶終了ボタン”かどうかを調べる。ステップS210で押されたボタンが“記憶終了ボタン”であればアクションモードをクリアして、一連のキー操作の取り込みを中止させ(ステップS220)、バッファの内容に該当ないイベント名を付けてイベントテーブル50へ格納し(ステップS230)、次のイベントをセンシングするためにステップS110へ戻る。“記憶終了ボタン”からの信号でなければステップS240へ進む。ステップS240では、押されたボタンが“イベント一覧ボタン”かどうかを調べる。ステップS240で押されたボタンが“イベント一覧ボタン”でなければ、“終了ボタン”が押されたものとして、本システムの実行を終了する。ステップS240で押されたボタンが“イベント一覧ボタン”

であれば、イベント一覧表を表示する。この中からバッファ内の文字列に対して付けたイベント名に割り付けるイベントを選択する(ステップS250)。選択されたイベントとバッファ内の文字列に与えたイベント名とを対応付けてマッピングテーブル60へ格納し(ステップS260)、次のイベントをセレクトするためにステップS110へ戻る。

【0013】次にイベント取込手段の処理を説明する。上述したステップ250で、どのようなキーアクションのときそのイベント(アクション取込手段20で設定したイベント名とアクション列)を引き起こすかをユーザ自身によって設定するために、“ユーザによる設定”を選択することもできる。これ以降の処理は、図7で示すイベント取込手段40でなされるが、処理の方法は、ほぼアクション取込手段の説明と類似している。まず、実行に先立ち一連のキー操作を取り込んでいる状態(以下、アクションモードという。)か、そうでないかを示すモードスイッチをクリアする(ステップS300)。キーボードおよびマウス等からの入力信号がいかどうかをセレクトする(ステップS310)。このセレクトした結果、“記憶開始ボタン”、“記憶終了ボタン”または“終了ボタン”等の“ボタン”からのイベントであるかどうかを調べる(ステップS320)。ボタンからの信号の時にはステップS330へ進む。そうでなければステップS360へ進む。“記憶開始ボタン”からのイベントかどうかを調べる。そうであればステップS340へ進む。そうでなければステップS410へ進む。“記憶開始ボタン”からの信号の時には、キー操作列を文字列化して一時的に保持するイベントバッファをクリアする(ステップS340)。以後、アクションモードとして設定し、キー操作列を取り込むようにして(ステップS350)、ステップS370へ進む。ステップS320で“ボタン”からのイベントでなかった場合、キー又はマウス等からの入力信号であるとする。このとき既にアクションモードに設定されているかを調べる(ステップS360)。まだ、アクションモードに設定されていないければ、このキー操作を取り込まないので、次のイベントをセレクトするためにステップS310へ戻る。既にアクションモードになっているときには、ステップS370へ進む。

【0014】入力されたキー信号を文字列に変換し、その文字列をさらに“”で囲んだ文字列にする。この文字列をイベントバッファへ追加する(ステップS370)。また、イベントバッファが空の状態でないときには、文字カンマ(,)を付けて上記の文字列をイベントバッファに追加する。今処理しているキーと同時に別のキーも押されているかを調べる(ステップS380)。押されているればステップS390へ進む。押されておらずすべてのキーがアップの状態になっているれば、次のイベントセレクトのためにステップS310へ戻る。同時

に、複数のキーが押されていれば、文字プラス(+)をバッファへ追加し(ステップS390)、その別のキー信号を文字列に変換し、その文字列をさらに“”で囲んだ文字列をバッファへ追加し(ステップS400)、別のキーが同時に押されているかを調べるためにステップS380へ戻る。ステップS330で“記憶開始ボタン”でないと判断されたとき、即ち“記憶終了ボタン”からの信号であるとして、イベントバッファに保持された文字列を実際にイベントを引き起こすキー操作として、イベント名(このイベント取込手段40を起動したときに取り扱っていたバッファ内の文字列に与えたイベント名)とここで取り込んだ文字列(イベントを引き起こすトリガーとなるアクション列)とをマッピングテーブル60へ格納し(ステップS410)、イベント取込手段40の処理を終了する。

【0015】本発明は、上記の実施形態のみに限定されたものではなく、例えば、図8に示したハードウェア構成を持つコンピュータ100によっても実現が可能である。図8に示すように、コンピュータ100は、入力装置101と、表示装置102と、CPU(Central Processing Unit; 中央処理ユニット)103と、メモリ104と、記憶装置105と、媒体駆動装置106およびこれらを接続するバス107とを有している。入力装置101は、キーボード、マウス、タッチパッド等によって構成され、情報の入力に使用される。表示装置102は、種々の出力情報や入力装置1からの入力された情報などを表示出力させる。CPU103は、種々のプログラムを動作させる。メモリ104は、プログラム自身を保持し、又そのプログラムがCPU103によって実行されるときに一時的に作成される情報等を保持する。記憶装置105は、本システムで扱うイベントテーブル50と、マッピングテーブル60およびプログラムとプログラム実行時の一時的な情報等を保持する。媒体駆動装置106は、プログラムやデータ等を記憶した記録媒体を装着してそれらを読み込み、メモリ104または記憶装置105へ格納するのに用いられる。又、媒体駆動装置106は、直接データの入出力やプログラム実行するのに使ってもよい。図1に示したキー割当てシステムを構成する各機能を実施するためのプログラムを作成する。このプログラムを予めCD-ROM等の記録媒体に書き込んでおき、このCD-ROMを各サイトのCD-ROMドライブのような媒体駆動装置6を搭載したコンピュータ100に装着して、前記プログラムを読み取ってメモリ104または記憶装置105に格納し、前記プログラムを実行することによって、上記の実施形態と同様の機能を実現することができる。尚、記録媒体としては半導体媒体(例えば、ROMまたはICメモリカード等)、光媒体(例えば、DVD-ROM、MO、MDまたはCD-R等)、磁気媒体(例えば、磁気テープまたはフレキシブルディスク等)のいずれであってもよい。

12

10 制御手段、20 アクション取込手段、30 イベント割当手段、40 イベント取込手段、50 イベントテーブル、60 マッピングテーブル、100 コンピュータ、101 入力装置、102 表示装置、103 CPU、104 メモリ、105 記憶装置、106 媒体駆動装置、107 バス。

【発明の効果】以上説明したように、請求項1および4の発明によれば、ユーザーによる操作フロー、キー操作として記憶することができると、従来、複合のキー操作が必要であったものが特定のキー、マウス操作により実現でき、作業効率を上げることができると。請求項1および4の発明によれば、例えば、「パソコンの電源をOFFにする。」および「定型的なメールを送る際に、メーリングソフトを立ち上げ、メールを書き、宛先を指定し、メールを送信し、メーリングソフトを閉じる。」といったことをワンアクションで行うことができる。また、請求項2および3の発明によれば、キー操作を文字列としてとにより、実際にキーを操作するだけでなく、取り込まれたキー操作列としての文字列だけでなく、直接文字列として入力されたキー操作列を編集することができるので、キーへの割当て可能な格納位置とすることができると。また、請求項3および6の発明によれば、一連のキー操作を引き起こせるイベントを決定し起こせるトリガーとなる一連のキー操作をユーザが設定できるように、小回りのきくシステムを作成できる。請求項3および6の発明によれば、例えば、「パスワードを入力する」とことによりある機密の作業（「パスワードを入力する」ということ）を

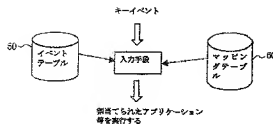
【圖3】



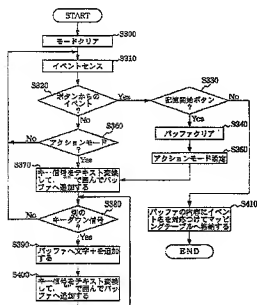
【圖 4】



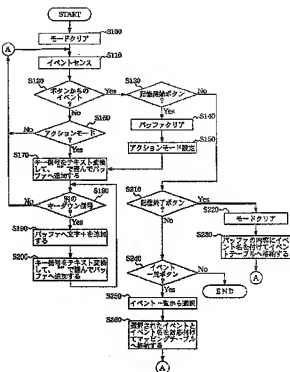
【図5】



【図7】



【図6】



【図8】

